Available online at www.sciencedirect.com

ScienceDirect

Journal of Computer and System Sciences 73 (2007) 1121–1130

JOURNAL OF COMPUTER AND SYSTEM SCIENCES

www.elsevier.com/locate/jcss

A new approach to model virtual channels in interconnection networks

N. Alzeidi [a,*], A. Khonsari [b], M. Ould-Khaoua [a], L. Mackenzie [a]

[a] *Department of Computing Science, University of Glasgow, Glasgow, UK*

[b] *Department of ECE, University of Tehran, School of Computer Science, Institute for Studies in Theoretical Physics and Mathematics, Tehran, Iran*

Received 3 October 2005; received in revised form 11 March 2006

Available online 24 February 2007

Abstract

Dealing with virtual channels has always been a critical issue in developing analytical performance models for interconnection networks. Almost all previous studies relied on a method proposed by Dally to capture the effect of virtual channels multiplexing in the performance of interconnection networks. This paper presents a new method to model the effect of virtual channel multiplexing in high-speed wormhole-switched interconnection networks. Dally's method loses its accuracy as the traffic load increases due to blocking nature of wormhole-switched networks. Our new method is based on a finite capacity queue, $M/G/1/V$ and comparing to Dally's method achieves a higher degree of accuracy under low, moderate and high traffic loads. Furthermore, its simplicity eases its employment under different network conditions and setup. The presented model is validated by means of an event driven simulator and a detailed comparison with Dally's method is presented.

Keywords: Analytical models; Interconnection networks; Virtual channels; Wormhole switching

1. Introduction

Wormhole switching [8] has become the dominating switching technique used in contemporary multicomputers and more recently in clusters [9] and system area networks [15]. This is because it requires minimum buffer space and it makes message latency almost independent of the message distance in absence of blocking. In wormhole switching, a message is broken into flits (few bytes each) for transmission and flow control. The header flit (the only flit that contains routing information) governs the route and the remaining data flits follow in a pipelined fashion. If the header flit is blocked, the data flits are blocked in situ.

As network traffic increase, messages may experience large delay to cross the network due to chains of blocked channels. To reduce the blocking delay, the flit buffers associated with a given physical channel are organized into several virtual channels [3], each representing a "logical" channel with its own buffer and flow control logic. Virtual

* Corresponding author.

E-mail addresses: zeidi@dcs.gla.ac.uk (N. Alzeidi), ak@ipm.ir (A. Khonsari), mohamed@dcs.gla.ac.uk (M. Ould-Khaoua), lewis@dcs.gla.ac.uk (L. Mackenzie).

0022-0000/$ – see front matter
doi:10.1016/j.jcss.2007.02.002

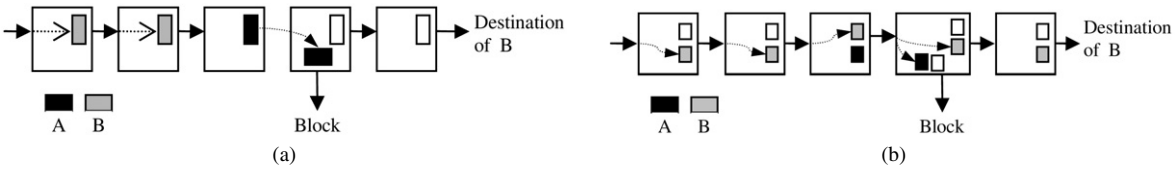


Fig. 1. (a) Message B is blocked behind message A, while physical channels remain idle. (b) Virtual channels provide additional buffers allowing message B to pass blocked message A.

channels are allocated independently to different messages and compete with each other for the physical bandwidth. This de-coupling allows messages to bypass each other in the event of blocking, using network bandwidth that would otherwise be wasted. Adding virtual channels to wormhole-switched networks greatly improves performance because they reduce blocking by acting as "bypass" lanes for non-blocked messages. Figure 1 illustrates the use of virtual channels as bypass lanes.

The concept of virtual channels has also been exploited to develop deadlock-free routing algorithms [4]. A routing algorithm specifies how a message selects its network path. Dealing with deadlock situations, that is when no message can advance towards its destination due to blocked channels, is a critical requirement for any routing algorithm. Deterministic routing [8] has been widely deployed in existing multicomputers because it is simple to implement and requires minimum number of virtual channels [10,16,17]. However, messages with the same source and destination addresses always follow the same path.

As a result, they cannot take advantage of the alternative paths that a topology may provide to reduce latency and avoid faulty links. In the other hand, many adaptive routing algorithms have been proposed where a message can use any of the available alternative paths between a given pair of nods to advance towards its destinations [6,7].

Many analytical models have been proposed to evaluate the performance merits of different routing algorithms in multicomputers [2,11–13,18–20]. Almost all of these models have used a method proposed by Dally [3] to capture the effect of virtual channel multiplexing in the network. The method proposed by Dally is based on a Markov process and although it is useful in some cases, it loses its accuracy as network traffic increases. This paper proposes a new general method for modeling virtual channel multiplexing based on finite capacity $M/G/1/V$ queuing system. The new method can easily be tailored for different traffic conditions by simply using different service time distributions.

The rest of this paper is organized as follows. Section 2 briefly explains Dally's method of virtual channel multiplexing and shows its relation to $M/M/1$ queuing system. Section 3 is devoted to development of the new general model. In Section 4 we validated the new model and presented some experimental results. Finally, we conclude the study in Section 5 and present some future directions.

2. Dally's method

In Dally's original paper [3], the average degree of virtual channel multiplexing (the multiplexing factor), $\overline{V}$ is given by Eq. (9) in [3] and is equal to

$$\overline{V} = \frac{\sum_{v=0}^{V} v^2 p_v}{\sum_{v=0}^{V} v p_v}, \tag{1}$$

where V is the total number of virtual channel used per physical channel and P_v is the probability of v virtual channels being busy. Dally determined P_v using a Markov process [3], shown in Fig. 2. State V_v corresponds to v virtual channels being busy. The transition rate out of state V_v to state V_{v+1} is λ_c where λ_c is the traffic rate on each channel, while the rate out of state V_{v+1} to state V_v is $1/\overline{S}$, where $\overline{S}$ is the mean service time. The transition rate out of the last state is reduced by λ_c to account for the arrival of messages while a channel is in this state. Solving this model for the steady state probabilities gives

$$q_v = \begin{cases} 1, & v = 0, \\ q_{v-1}\lambda_c \overline{S}, & 0 < v < V, \\ q_{v-1}\frac{\lambda_c}{1/\overline{S}-\lambda_c}, & v = V, \end{cases} \tag{2}$$

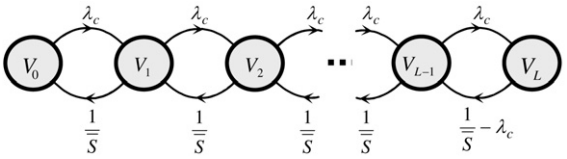


Fig. 2. Markov model for virtual channel occupancy.

$$P_v = \begin{cases} \frac{1}{\sum_{v=0}^{V} q_v}, & v = 0, \\ P_{v-1}\lambda_c \overline{S}, & 0 < v < V, \\ P_{v-1}\frac{\lambda_c}{1/\overline{S}-\lambda_c}, & v = V. \end{cases} \tag{3}$$

We now present the following lemma.

Lemma. *P_v as calculated by Dally in* [3] *is exactly the probability of the number of customers in an $M/M/1$ queuing system.*

Proof. Rewriting (3) yields the following equations:

$$P_0 = \frac{1}{\sum_{v=0}^{V} q_v} = \frac{1}{q_0 + \lambda_c \overline{S} q_0 + \cdots + (\lambda_c \overline{S})^{v-1} q_0 + (\lambda_c \overline{S})^{v-1} q_0 \frac{\lambda_c}{1/\overline{S}-\lambda_c}} = \frac{1}{1 + \lambda_c \overline{S} + \cdots + (\lambda_c \overline{S})^{v-1} + \frac{(\lambda_c \overline{S})^v}{1-\lambda_c \overline{S}}}$$

$$= \frac{1}{\frac{1-(\lambda_c \overline{S})^v}{1-\lambda_c \overline{S}} + \frac{(\lambda_c \overline{S})^v}{1-\lambda_c \overline{S}}} = 1 - \lambda_c \overline{S}. \tag{4}$$

Substituting this into (2) gives

$$P_v = \begin{cases} P_0 (\lambda_c \overline{S})^{v-1}, & 0 < v < V, \\ (1 - \lambda_c \overline{S})(\lambda_c \overline{S})^{v-1} \frac{\lambda_c}{1/\overline{S}-\lambda_c}, & v = V. \end{cases} \tag{5}$$

After some manipulation of the above equations, P_v can be rewritten as

$$P_v = \begin{cases} (1 - \lambda_c \overline{S})(\lambda_c \overline{S})^{v-1}, & 0 < v < V, \\ (\lambda_c \overline{S})^v, & v = V. \end{cases} \tag{6}$$

This is exactly the probability of the number of customers in an $M/M/1$ queuing system and hence the above lemma is proven. □

By virtue of the above lemma, Dally's method of calculating the probability of busy virtual channels is now reduced to the calculation of the number of customers in an $M/M/1$ queuing system. This approach is accurate under low traffic where the performance measures of $M/M/1$ queue do not deviate too much compared to other queues. However, as the traffic increases the blocking nature of the wormhole-switched networks interrupts the service time at each switch and the service becomes more general rather than exponential as in the $M/M/1$ queuing system. This explains the degradation of the accuracy of the analytical models that are based on Dally's method under moderate and high traffic.

3. A new general method

In this section we propose a new general method for calculating P_v. The probability, P_v, that v virtual channels at a given physical channel are busy, is the same across all network channels, and can be determined using the distribution of the number of customers in an $M/G/1/V$ queuing system, $\{P_v;\ 0 \leqslant v \leqslant V\}$. The arrival rate is assumed to be exponentially distributed with a mean of λ_c messages received by each physical channel and the service time is generally distributed. The queue size is set to be equal to the number of virtual channels per physical channel, V. Appendix A summarizes the notation used in the model. The fact that the service time is general gives us the freedom

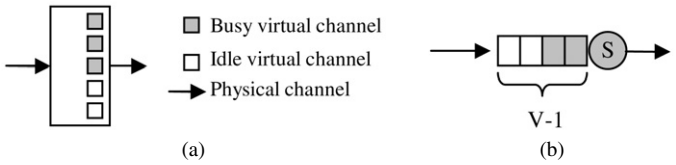


Fig. 3. (a) Three busy virtual channels corresponds to (b) Three customers in the $M/G/1/V$ queuing system: two in the queue and one being serviced.

to either assume will-known service time distributions or to approximate it using approximation methods. This gives the model the flexibility to be adapted to different network and traffic conditions and hence more accurate results.

As illustrated in Fig. 3, when there are v customers in the system this corresponds to v virtual channels being requested. The probability that v virtual channels are busy, for $(0 \leqslant v \leqslant V)$, is the probability of v customers in the system (including the one on service). Hence we can write

$$P_v = \Pr[N = v], \quad 0 \leqslant v \leqslant V. \tag{7}$$

For the paper to be self contained we now briefly present the derivation of the distribution of the number of customers in $M/G/1/V$ queuing system according to Takagi [21]. The distribution $\{P_v;\ 0 \leqslant v \leqslant V\}$ of the queue size at an arbitrary time can be obtained from the distribution $\{\pi_v;\ 0 \leqslant v \leqslant V-1\}$ of the queue size embedded at service completion points by the method of semi-Markov process. Hence, the distribution $\{P_v;\ 0 \leqslant v \leqslant V\}$ of the queue size at arbitrary time is given by

$$P_v = \begin{cases} \frac{\pi_0}{\pi_0+\rho}, & v = 0, \\ (1-P_B)\pi_v, & 1 \leqslant v \leqslant V-1, \\ (1-P_B)(\rho+\pi_0-1), & v = V, \end{cases} \tag{8}$$

where $\rho = \lambda_c \overline{S}$ is the offered load and the blocking probability, P_B that an arriving customer is blocked from joining the queue because the system is full is given by

$$P_B = 1 - \frac{1-P_0}{\rho}. \tag{9}$$

An efficient algorithm for computing $\{\pi_v;\ 0 \leqslant v \leqslant V-1\}$ can be given in terms of

$$\pi'_v \triangleq \frac{\pi_v}{\pi_0}, \quad 0 \leqslant v \leqslant V-1, \tag{10}$$

where $\{\pi'_v;\ 0 \leqslant v \leqslant V-1\}$ can be recursively calculated as follows

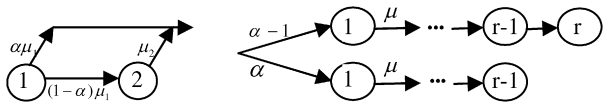
$$\pi'_0 = 1, \tag{11}$$

$$\pi'_{v+1} = \frac{1}{a_0}\left(\pi'_v - \sum_{j=1}^{v} \pi'_j a_{v-j+1} - a_v\right), \quad 0 \leqslant v \leqslant V-2, \tag{12}$$

$$\pi_0 = \left(\sum_{v=0}^{V-1} \pi'_v\right)^{-1}, \tag{13}$$

where $a_v = \int_0^\infty \frac{(\lambda_c x)^v}{v!} e^{-\lambda_c x} f_S(x)\,dx$, $v = 0, 1, 2, \ldots$, is the probability that v customers arrive during the service time which is expressed as the probability density function (pdf), $f_S(x)$, in the above equation.

The important thing about our approach is the simplicity of adapting it to fit different network conditions and traffic loads by only changing the pdf of the service time in Eq. (14). However, when the knowledge of the service time distribution is not available, we still can use the new approach straightforwardly by trying to approximate the service time distribution. Kleinrock [14] has shown that any distribution function can be approximated as closely as desired by a series-parallel stage-type of exponential distributions. This approach has been widely used to approximate the distribution of complex functions that knowledge of them is not available. We propose the following approximation based on the squared coefficient of variation of the service time, cv^2.



(a) (b)

Fig. 4. Phase diagram for (a) two-phase Coxian distribution ($Cox-2$) and (b) mixed Erlang distribution ($E_{k,k-1}$).

When ($cv^2 \geqslant 1/2$), the service time can be approximated by using a two-phase Coxian distribution ($Cox-2$). If, in the other hand, ($cv^2 < 1/2$), we match the first two moments by using a mixed Erlang distribution ($E_{k,k-1}$). Figure 4 illustrates the phase diagrams for both distributions. Therefore we can express the Laplace transform of the approximated service time distribution as

$$S^*(s) = \begin{cases} \frac{\alpha\mu_1(\mu_2+s)+(1-\alpha)\mu_1\mu_2}{(\mu_1+s)(\mu_2+s)}, & cv^2 \geqslant 0.5, \\ \frac{s\alpha\mu^{r-1}+\mu^r}{(\mu+s)^r}, & cv^2 < 0.5. \end{cases} \tag{14}$$

The parameters of the $Cox-2$ distribution (α, μ_1, μ_2) and the $E_{k,k-1}$ distribution (α, μ, r), are calculated so that the first two moments of the approximated distribution matches those of the actual service time distribution. Matching the moments is done by finding the first and second derivatives of the Laplace transform and setting s to 0. Hence the parameters for the $Cox-2$ distribution are given by

$$\begin{aligned} &\alpha = 1 - \frac{1}{2\,cv^2}, \\ &\mu_1 = 2S, \\ &\mu_2 = \mu_1\alpha. \end{aligned} \tag{15}$$

Similarly the parameters for the $E_{k,k-1}$ distribution can be written as

$$\begin{aligned} &\alpha = \frac{1}{1+cv^2}\left[rcv^2 - \sqrt{r\left(1-cv^2\right) - r^2cv^2}\right], \\ &\mu = \frac{r-\alpha}{S}, \\ &r = \left\lceil \frac{1}{cv^2} \right\rceil. \end{aligned} \tag{16}$$

Now we have an approximated distribution for the service time that can be used in (14) and hence the probability of busy virtual channels is calculated using (8).

It should be mentioned that, the two moments matching requires knowledge of the variance of the service time, σ_S^2. Finding the exact expression of σ_S^2 is a difficult and complex task since this depends on several parameters. Furthermore, given that we are driven by the requirement for analytic simplicity as well as the desire of versatility and practicality, we can use the simple approximation suggested by Draper and Ghosh [5] for computing the variance of the service time. Since the minimum service time at a given channel is equal to the message length, M, the variance of the service time can be approximated as $\sigma_S^2 = (\overline{S} - M)^2$.

4. Experimental results

In this section we validate our new method of calculating the probabilities of busy virtual channels. The validation process is two-fold. We first validate the calculation of the probabilities of the busy virtual channels predicted by our method and Dally's method compared to the results obtained by an event driven simulator. Second, we deploy our new method of calculating the probabilities of busy virtual channels into an analytical model that have previously used Dally's method to calculate the virtual channel multiplexing [18]. For the first validation step, we plot the probabilities of busy virtual channels for low, moderate and high traffic conditions. The probabilities obtained by our method and by Dally's method are plotted against an event-driven simulator. The simulator mimics the behavior of a two dimensional unidirectional wormhole-switched torus. Figure 5 depicts the probability of busy virtual channels predicted by both models (labeled as Dally and General in the figures) plotted against those provided by the simulator (labeled as

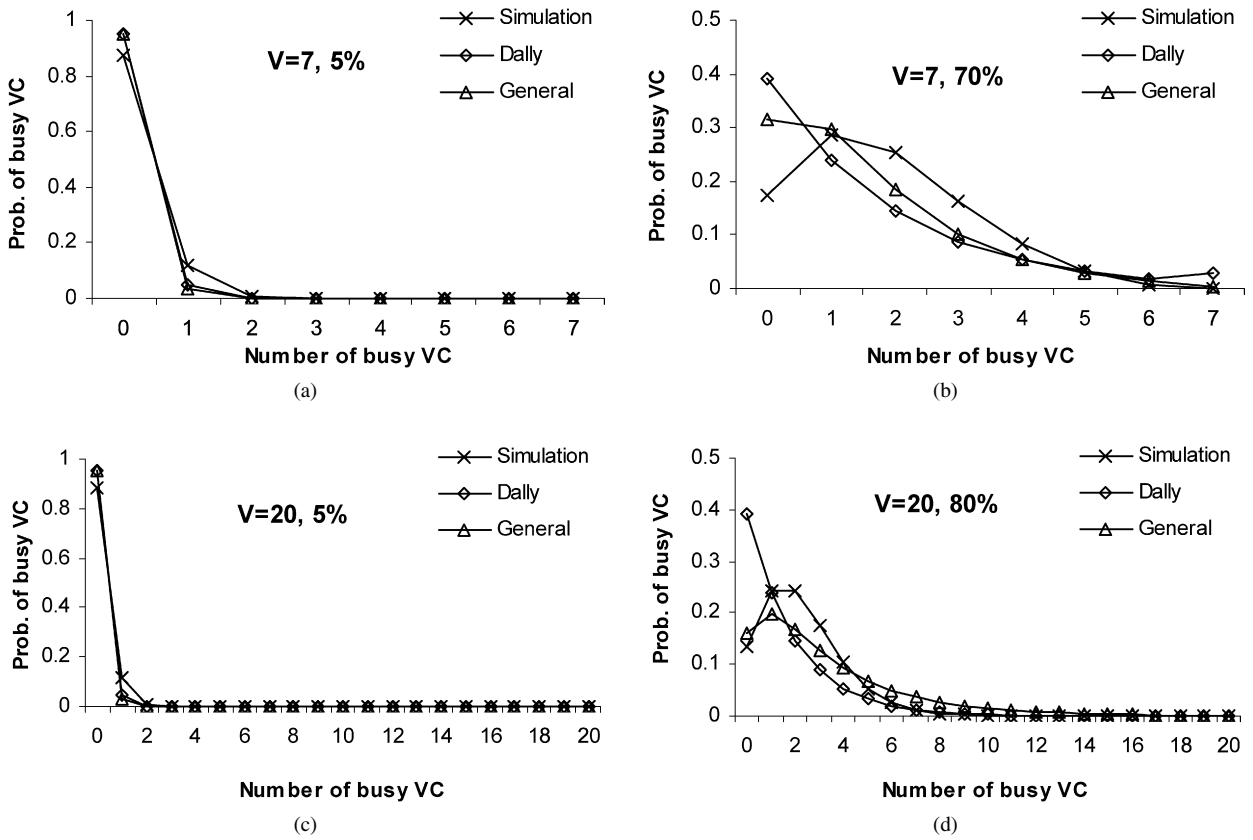


Fig. 5. Probability of busy virtual channels as predicted by Dally's method and by the new general method plotted against the probabilities obtained from an event-driven simulator: (a) $V = 7$ and server utilization is 5%, (b) $V = 7$ and server utilization is 70%, (c) $V = 20$ and server utilization is 5%, (d) $V = 20$ and server utilization is 80%.

Simulation in the figures) as a function of the number of virtual channels used in the 8-ary 2-cube. As can be seen from the figure, under light traffic (utilization 5%) both models seem to agree with the results obtained by the simulator. As we explained earlier this is an expected behavior as the network is not heavily loaded and hence the effect of blocking is not noticeable. In other words, under low traffic it is still acceptable to approximate the service time with an exponential distribution. However, as the traffic increases (utilization 70%), the discrepancy between the simulator results and Dally's method becomes more apparent. Meanwhile, the results predicted by our new method stay in close agreement with those results obtained by the event-driven simulator. Again this is due to the high blocking encountered by wormhole switched networks under moderate and high traffic conditions.

Our second validation step is to deploy our new method into an analytical model that has previously used Dally's method to capture the effect of virtual channel multiplexing. There have been many analytical models in the literature that employed Dally's method to capture the effect of virtual channel multiplexing. In [18], Ould-Khaoua presented an accurate analytical model for Duato's fully adaptive routing algorithm [6] and employed the Dally's method [3] for virtual channel multiplexing. For the purpose of illustration and comparison, in this section, we will amend the model presented in [18] with our new approach and compare it with Dally's method. The implementations of both methods are identical except for the calculation of the probability of busy virtual channels. This is to make sure that the differences are due to the different methods of calculating the degree of the virtual channel multiplexing. It is noteworthy to mention that selecting different latency models should have no effect in the relative comparison of the virtual channels models as the calculation of the probability of busy virtual channels using the general method makes no assumptions in regard to the underlying latency model. Nevertheless, we have experimented with different latency models and more results can be found in [1].

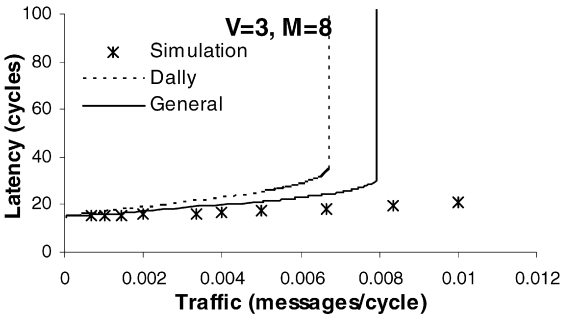
Appendix A contains a list of all the notations used in the model. Furthermore, we will keep the same assumptions adopted in [18], which are outlined below for the purpose of completeness

(1) Message destinations are uniformly distributed across the network nodes.
(2) Nodes generate traffic independently of each other and according to a Poisson process with mean rate of λ_g messages per cycle.
(3) Message length is fixed at M flits, each of which requires one cycle to be transmitted from one router to another. Moreover, a message is long enough so that its data flits span from the source to the destination.
(4) The local queue at the injection channel in the source node has infinite capacity. Messages at the destination node are transferred to the local PE as they arrive.
(5) V ($V > 2$) virtual channel are used per physical channel. In Duato's routing algorithm [6], class a contains $(V-2)$ virtual channels, which are crossed adaptively, and class b contains two virtual channel, which are crossed deterministically (e.g. in an increasing order of dimensions). Let the channels in class a and b called adaptive and deterministic virtual channel respectively. When there is more than one adaptive virtual channel available, a message chooses one at random. Even though there are two deterministic virtual channels, a message can use only one at a time.

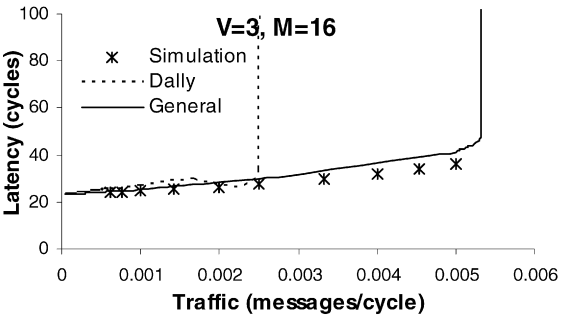
As we mention earlier, we keep the same implementation for both models except for the calculation of the probability of busy virtual channels. The model in [18] calculate the probability of busy virtual channels based on Dally's method by using (2) and (3) or alternatively, as we showed, by using (6). The alternation we made is to use the new general model based on the $M/G/1/V$ queuing system to calculate the probability of busy virtual channels using (7). We experimented with different service time distributions. The results presented in this paper are based on Erlang service time distribution. Our new model and Dally's model are both plotted against results obtained from an event-driven simulator that mimics the behavior of a unidirectional wormhole-switched k-ary n-cube with multiple virtual channels. The network cycle time is defined as the transmission time of a single flit from one router to the next. Processors at all nods generate fixed size messages (M flits each) randomly with an exponential distribution of inter-arrival time. Destination nodes are determined using a uniform random number generator. The mean message latency is defined as the mean amount of time from the generation of a message until the last data flit reaches the local PE at the destination node. In each simulation experiment, a total number of 100 000 messages are delivered to their destinations. To avoid the distortions due to start-up conditions, the first 10 000 messages are ignored. Numerous validation experiments have been performed for several combinations of network sizes, message lengths, and virtual channels to assess the accuracy of the proposed analytical model. However for the sake of specific illustration, latency results are presented for unidirectional 8-ary 2-cube interconnection network. Virtual channels takes the values of 3, 5, 7 and 20 per physical channel and the message length is kept fixed at $M = 8$ and $M = 16$.

Figure 6 depict the mean message latency results predicted by both models (Dally and General) plotted against those provided by the simulator (Simulation) as a function of the traffic injected in the 8-ary 2-cube. The figure reveals that both the models that are based on Dally's method and our new general method are in close agreement with simulation results under low traffic. However, as the injection rate increases, the model based on Dally's approach starts to deviate from the simulation results. Meanwhile, the model that is based on our new general approach continues to match the simulation results as the network approaches the saturation point. However, some discrepancies are still apparent due to the approximations made to ease the derivation of the model. Namely, the approximation we made to determine the variance and the approximation of the service time as Erlang distribution. Nevertheless, it can be concluded that our new general approach of calculating the probability of busy virtual channels is, in one hand, more accurate than Dally's method under different traffic loads, and on the other hand, it can be customized to fit different traffic and network conditions by using different distributions for the service time.

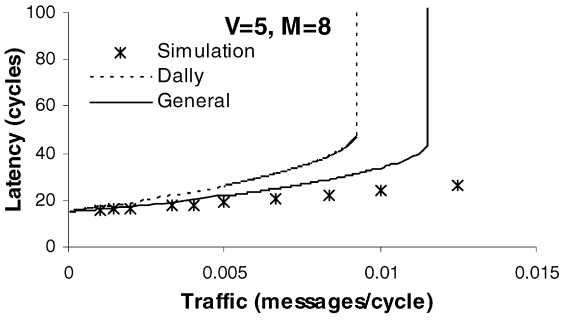
To conclude this section we should mention that an insight investigation of the model equations reveals that the complexity of the new general method and Dally's method are indeed of the same order. Recall that Dally's method calculates the probability of busy virtual channels using (2) and (3). For V virtual channels, it can be noticed that the computation of the probability that non of the virtual channels is busy, P_0, using Dally's method is of order $O(V)$, while the computation of v busy virtual channels, where $v = 1, 2, \ldots, V$ is of order $O(1)$. In the other hand, the new general method calculates the probability of busy virtual channels using (7). The complexity of the new general method mainly resides in the calculation of the integral that represents the number of arrivals during the



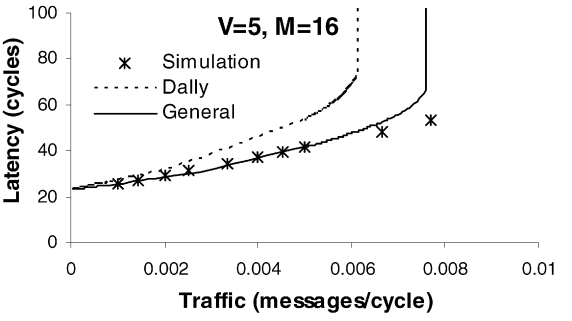
(a)



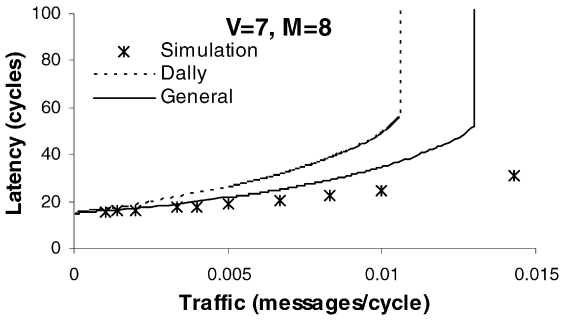
(b)



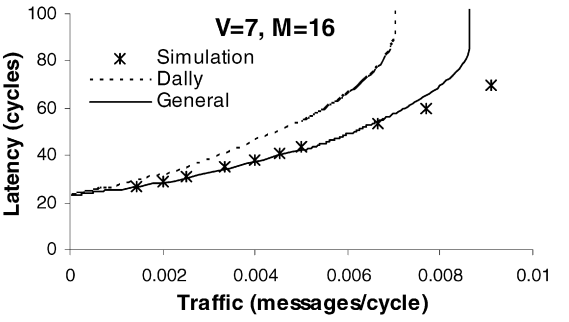
(c)



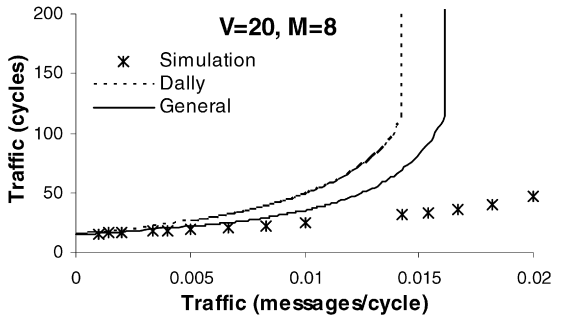
(d)



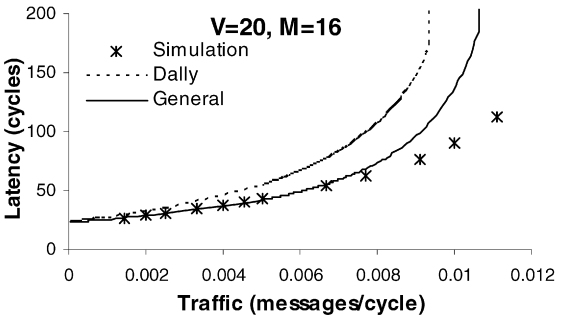
(e)



(f)



(g)



(h)

Fig. 6. Validation of the model and comparison with Dally's method for 8-ary 2-cube interconnection network: (a) $V = 3$ and $M = 8$, (b) $V = 3$ and $M = 16$, (c) $V = 5$ and $M = 8$, (d) $V = 5$ and $M = 16$, (e) $V = 7$ and $M = 8$, (f) $V = 7$ and $M = 16$, (g) $V = 20$ and $M = 8$, (h) $V = 20$ and $M = 16$.

service $a_v = \int_0^\infty \frac{(\lambda_c x)^v}{v!} e^{-\lambda_c x} f_S(x)\,dx$. Numerical methods can be used to compute the above integral. However, the complexity of the new model can be minimized by calculating the above integral offline and feeding its results to the new general method as an input. In this case the calculation of the probability of busy virtual channels using the new general method is in the same order as Dally's method.

5. Conclusions

Almost all previous analytical models developed to assess the performance of wormhole-switched networks, employed a method presented by Dally. Dally's method is accurate only under light traffic and degrades significantly as the traffic increases. In this study, we proposed a new general model to capture the effect of virtual channel multiplexing. Our model is based on an $M/G/1/V$ queue instead of a Markov process as opposed to Dally's. We showed that Dally's method is equivalent to finding the queue size distribution of an $M/M/1$ queuing system. This explains the accuracy degradation of analytical models based on Dally's approach especially under moderate and high traffic. Beside the accuracy that it achieves under low, moderate and high traffic, a main advantage for our new approach is also the simplicity of adapting it to work with different traffic conditions and network setups by changing the pdf of the service time distribution. Furthermore, a two-moment approximation is presented to be used when only the first two moments of the service time is available.

The validity and accuracy of the model has been demonstrated by comparing it to Dally's method as will as to a discrete event simulator. Results show that our model is more accurate than Dally's especially under moderate and high traffic. Furthermore, the design of the model is so general so that it can be tailored to fit different traffic conditions and network topologies. Future research will focus on setting criteria to define what type of service time distributions to use under different traffic conditions and network setups. We will also be investigating the possibilities of approximating the variance of the service time more accurately.

Appendix A. Notation used in this paper

Symbol	Description
λ_g	Message generation rate at a node
λ_c	Traffic rate received by each physical channel
$\overline{V}$	Degree of virtual channel multiplexing
V	Total number of virtual channel per physical channel
P_v	Probability of v busy virtual channels
$\overline{S}$	Mean message latency
q_v	Intermediate variable used by Dally to calculate the probability of busy virtual channels
P_B	Blocking probability in $M/G/1/V$ queuing system
π_v	Probability of v customers in $M/G/1/V$ system after service completion
π'_v	Intermediate variable used to calculate π_v
ρ	Offered load in $M/G/1/V$ queuing system
a_v	Probability that v customers arrive during service in $M/G/1/V$ queue
$f_{\overline{S}}(x)$	The probability density function (pdf) of the service time
cv^2	Square coefficient of variation of the service time distribution
$S^*(s)$	Laplace transform of the service time distribution
$\sigma^2_{\overline{S}}$	Variance of the service time distribution
M	Message length

References

[1] N. Alzeidi, A. Khonsari, M. Ould-Khaoua, L.M. Mackenzie, A new queuing model for the analysis of virtual channels occupancy in wormhole-switched networks, Department of Computing Science, University of Glasgow, Glasgow, Technical Report TR-2005-206, 2005.

[2] Y. Boura, C.R. Das, T.M. Jacob, A performance model for adaptive routing in hypercubes, presented at International Workshop Parallel Processing, 1994.

[3] W.J. Dally, Virtual channel flow control, IEEE Trans. Parallel Distrib. Systems 3 (2) (1992) 194–205.

[4] W.J. Dally, C.L. Seitz, Deadlock-free message routing in multiprocessor interconnection networks, IEEE Trans. Comput. 36 (5) (1987) 547–553.

[5] J.T. Draper, J. Ghosh, A comprehensive analytical model for wormhole routing in multicomputer systems, J. Parallel Distrib. Comput. 23 (2) (1994) 202–214.

[6] J. Duato, A new theory of deadlock-free adaptive routing in wormhole networks, IEEE Trans. Parallel Distrib. Systems 4 (12) (1993) 1320–1331.
[7] J. Duato, P. Lopez, Performance evaluation of adaptive routing algorithms for k-ary n-cubes, presented at First International Workshop on Parallel Computer Routing and Communication, 1994.
[8] J. Duato, S. Yalamanchili, L.M. Ni, Interconnection Networks: An Engineering Approach, Morgan Kaufmann Publishers Inc., Los Alamitos, 2002.
[9] V. Halwan, F. Ozguner, A. Dogan, Routing in wormhole-switched clustered networks with applications to fault tolerance, IEEE Trans. Parallel Distrib. Systems 10 (10) (1999) 1001–1011.
[10] R.E. Kessler, J.L. Schwarzmeier, Cray t3d: A new dimension for cray research, in: 1993 IEEE Compcon Spring, no. 176, 1993.
[11] A. Khonsari, M. Ould-Khaoua, Compressionless wormhole routing: An analysis for hypercube with virtual channels, Comput. Electr. Eng. 30 (1) (2004) 45.
[12] A. Khonsari, H. Sarbazi-Azad, M. Ould-Khaoua, A performance model of software-based deadlock recovery routing algorithm in hypercubes, Parallel Process. Lett. 15 (1–2) (2005) 153.
[13] A. Khonsari, A. Shahrabi, M. Ould-Khaoua, H. Sarbazi-Azad, Performance comparison of deadlock recovery and deadlock avoidance routing algorithms in wormhole-switched networks, IEE Proc. Comput. Digital Techn. 150 (2) (2003) 97.
[14] L. Kleinrock, Queueing Systems, vol. 1, John Wiley, New York, 1975.
[15] S. Lee, Real-time wormhole channels, J. Parallel Distrib. Comput. 63 (3) (2003) 299–311.
[16] L.M. Ni, P.K. Mckinley, A survey of wormhole routing techniques in direct networks, Computer 26 (2) (1993) 62–76.
[17] M.D. Noakes, D.A. Wallach, W.J. Dally, J-machine multicomputer. An architectural evaluation, in: Conference Proceedings—Annual Symposium on Computer Architecture, no. 224, 1993.
[18] M. Ould-Khaoua, A performance model for Duato's fully adaptive routing algorithm in k-ary n-cubes, IEEE Trans. Comput. 48 (12) (1999) 1297–1304.
[19] H. Sarbazi-Azad, M. Ould-Khaoua, L.M. Mackenzie, An accurate analytical model of adaptive wormhole routing in k-ary n-cubes interconnection networks, Perform. Eval. 43 (2–3) (2001) 165–179.
[20] H. Sarbazi-Azad, M. Ould-Khaoua, A.Y. Zomaya, Design and performance of networks for super-, cluster-, and grid-computing: Part i, J. Parallel Distrib. Comput. 65 (10) (2005) 1119.
[21] H. Takagi, Queueing Analysis—A Foundation of Performance Evaluation, vol. 2, North-Holland, Amsterdam, 1993.